CICS/Transaction Server (z/OS)

ADS is distributed via our website (www.DebugCICS.com). All modules are pre-assembled for a specific release of CICS and are combined into a single object file, complete with all the necessary link-edit statements.

Source modules**:** The following source code is available for customization:

DSPEXITP is an exception table that contains the "generic" names of modules to be excluded from monitoring. You should review these entries to avoid conflicts with other system-oriented products and non-standard (custom) interfaces.

DSPNAMES is a table of pointers used to identify storage within the CICS region. You will increase the accuracy of the *ADS feedback* feature if you identify any additional control blocks you may have added to your system.

DSPSECTY is a security table that restricts user access to specific programs and files. It is intended for use in production or other sensitive environments where *ADS* would only be used by authorized personnel. The table can contain a list of generic (wild-card) terminal names or authorized operator id's.

DSPINITL is a macro that contains the default user profile and PF KEY definitions. By changing these parameters, *ADS* may not conform to the user manual.

Source Level Option:

This feature integrates your source code with the debugging process. For COBOL programs, it allows source statements to be dynamically inserted without the need for recompilation. This is accomplished with an additional jobstep that reads the compiler listing and stores the source reference information in a shared VSAM cluster.

Password Sign-on:

To activate *ADS*, run the **ADSM** transaction, which will prompt you for a password. If you don't have a password, or if your password has expired, *ADS* will display a four digit "parm code". To obtain a password contact us with the "parm code" via:

Telephone: 732 247-2727
Toll Free (US): 800 603-3114
Fax: 732 846-3515
Support@DebugCICS.com

CICS RESOURCE DEFINITIONS

TRANSACTIONS: (TASkDATAKey = **CICS,** TASKDAtaloc = **BELOW**)

```
TRansaction ==> ADSP          ADSM            SELE
PROGram     ==> DFHDSP        DSPENTRY        DSPDELET
PRIORity    ==> Ø             Ø               255
```

PROGRAMS: (Language = **Asm,** EXECKey = **CICS,** Datalocation = **BELOW**)

```
PROGram ==>DFHDSP     DSPENTRY   DSPDELET   DSPEDIT
           DSPFILE    DSPINITL   DSPNAMES   DSPROUTE
           DSPLABEL   DSPWHAT    DSP327Ø2   DSP327Ø5

RESident==>NO
THREADSAFE==NO


PROGram ==>DSPEXITP   DSPFMAIN   DSPTRACE   DSPICTRL

RESident==>YES  (required for these 4 programs)
```

FILES:

```
File         ==> ADSFILE
LSRPOOLID    ==> NONE
STRINGS      ==> 2
RECORDFormat ==> F
OPERATIONS   ==> ( all set ON )
```

CICS MULTI-REGION OPTION

To operate *ADS* in an MRO environment, the same set of PPT entries must be defined in each accessible region. In addition, the PCT entries must be defined as follows:

ADS requires the following three PCT entries:

ADSP	*ADS* Processor	(any four characters ending with "P")
ADSM	Menu Generator	(any four characters ending with "M")
SELE	Select Terminal	(any four character transaction id)

The PCT entries for ADSP and ADSM must be installed to allow the desired region to be accessed without the use of CRTE. For example:

Terminal region (TOR)	Application region (AOR-A)	Application region (AOR-B)
ADSP remote(AOR-A)	ADSP local	BDSP local
BDSP remote(AOR-B)		
TDSP local		
ADSM remote(AOR-A)	ADSM local	BDSM local
BDSM remote(AOR-B)		
TDSM local		

The "Select terminal" transaction must be fully cross-referenced to allow CICS to acquire and reroute the requested terminal for breakpoint processing.

Terminal region (TOR)	Application region (AOR-A)	Application region (AOR-B)
SELE remote(AOR-A)	SELE local	SELE remote(AOR-A)
BELE remote(AOR-B)	BELE remote(AOR-B)	BELE local
TELE local	TELE remote(TOR)	TELE remote(TOR)

INSTALLING THE FILES

Allocate a sequential file to receive the data.

```
//TEMP    EXEC  PGM=IEFBR14
//UPLOAD    DD  UNIT=SYSDA,DISP=(NEW,CATLG),
//              DCB=(BLKSIZE=8ØØ,LRECL=8Ø,RECFM=FB),
//              SPACE=(TRK,(50,10)),DSN=xxxxx.UPLOAD
```

Upload source tables: (example using `IND$FILE`)

```
C:\>SEND  IEBUPDTE.SRC  'xxxxx.UPLOAD' CRLF ASCII
```

Install source tables:

```
//TABLES  EXEC  PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD  SYSOUT=*
//SYSUT2    DD  DISP=OLD,DSN=user.source.library
//SYSIN     DD  DSN=xxxxx.UPLOAD,DISP=SHR
```

Upload the appropriate object file: (without conversion)

```
C:\>SEND  xxxxx.OBJ 'xxxxx.UPLOAD'
```

Install *ADS*:

```
//ADS     EXEC  PGM=IEWL,PARM='LIST'
//SYSPRINT DD  SYSOUT=*
//SYSLIB    DD  DISP=SHR,DSN=cics.loadlib       (for DFHEAI)
//SYSUT1    DD  UNIT=SYSDA,SPACE=(CYL,(4,1))
//SYSLMOD   DD  DISP=SHR,DSN=cics.user.loadlib
//SYSLIN    DD  DSN=xxxxx.UPLOAD,DISP=SHR
```

SOURCE LEVEL OPTION

1. Define the KSDS file with the IDCAMS utility.
2. Create a new CICS FCT entry for the dataset.
3. Insert //ADSFILE DD in the CICS jobstep JCL.
4. Build JCL procs for program compilation jobs.

IDCAMS:

```
DEFINE  CLUSTER -
        (NAME(vsam.adsfile) KEY(9 Ø) SHR(3,3) -
        CISZ(8192) RECORDSIZE(8182 8182) -
        RECORDS( number of programs * 1Ø ))
```

Initialize:

```
//INIT    EXEC PGM=ADSLABEL,PARM=DIR
//STEPLIB  DD  DISP=SHR,DSN=ads.loadlib
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  DUMMY
//SYSUT1   DD  DISP=SHR,DSN=vsam.adsfile
```

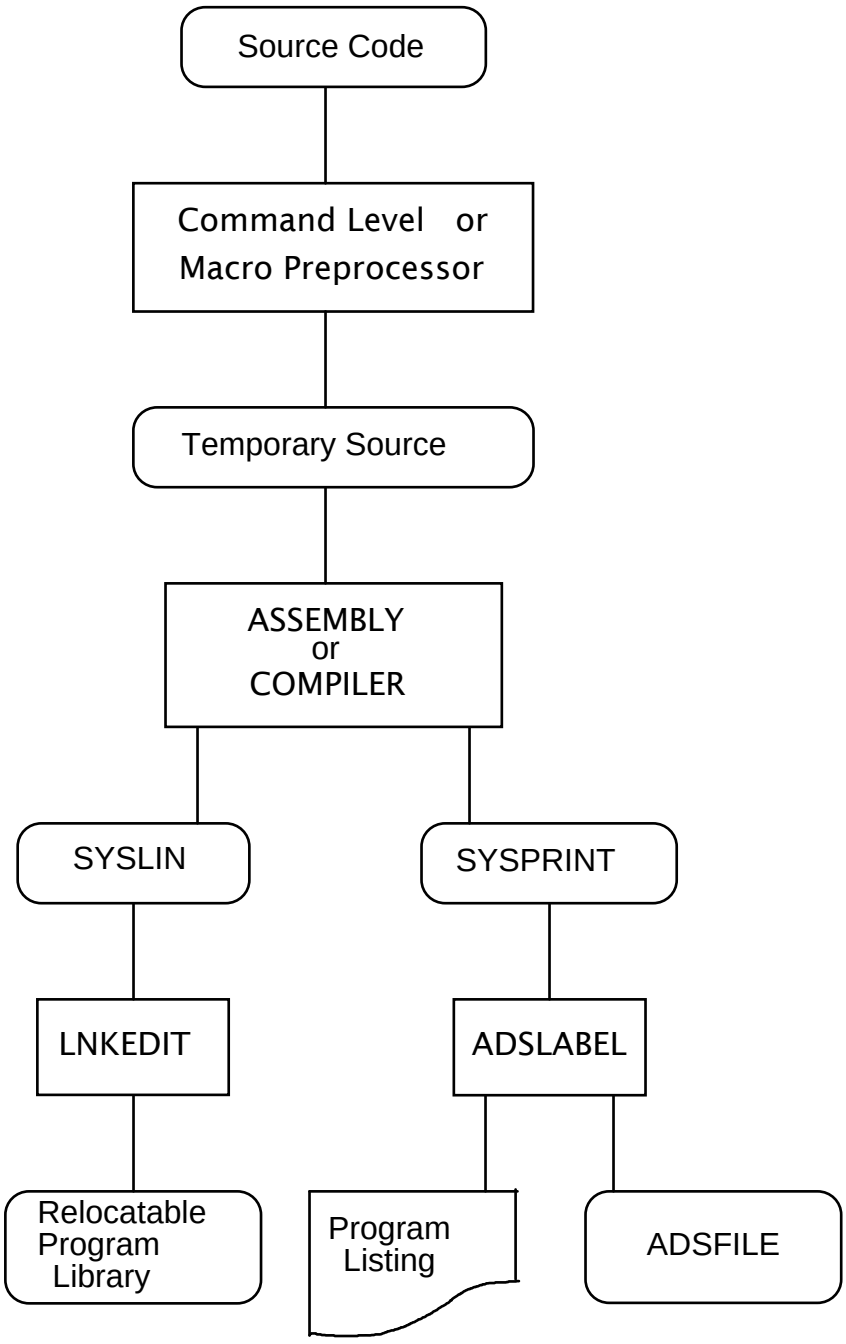
FCT:
(or CSD) Lsrpoolid(NONE)

```
DFHFCT  TYPE=DATASET,ACCMETH=VSAM,DATASET=ADSFILE,     C
        SERVREQ=(GET,BROWSE,UPDATE,NEWREC,DELETE),     C
        RECFORM=(FIXED,BLOCKED),STRNO=2,LSRPOOL=NONE
```

Do not use VSAM shared buffer resources for this dataset.

The ADSFILE is also used as the vehicle to maintain your installation's sign-on parameter.

SOURCE LEVEL PROCEDURE



COBOL PROCEDURE

```
//PRE     EXEC  pre-processor (Command or Macro Level)
//*
//COB     EXEC  PGM=cobol,
//              PARM=  to generate DMAP, PMAP, and VERB
//*             VS COBOL-II & UP  =  LIST, MAP, NOOFFSET
//SYSPRINT DD   UNIT=SYSDA,DISP=(MOD,PASS,DELETE),
//              SPACE=(TRK,(25,5)),DSN=&&LISTING,
//           DCB=(RECFM=FB,LRECL=121,BLKSIZE=2662) COBOL/CAPEX
//           DCB=(RECFM=FB,LRECL=133,BLKSIZE=266Ø) VS COBOL-II
//*
//ERRS    EXEC  PGM=ADSLABEL,PARM='DUMMY,NOPMAP,NODMAP',
//              COND=(5,GT,COB)          print cobol errors
//SYSPRINT DD   SYSOUT=*
//SYSIN     DD  DSN=&&LISTING,DISP=(OLD,DELETE)
//*
//ADS     EXEC  PGM=ADSLABEL,PARM='options',COND=(5,LT,COB)
//SYSPRINT DD   SYSOUT=*
//SYSIN     DD  DSN=&&LISTING,DISP=(OLD,DELETE)
//SYSUT1    DD  DISP=SHR,DSN=vsam.adsfile
//*
//LKED    EXEC  PGM=IEWL,COND=(5,LT,COB)
```

PARM *options*: (for ADS jobstep)

PGM=&*id*	-	Program PPT name (default is COBOL PROGRAM-ID).
NOSOURCE	-	Generates label reference information only. Saves disk space by not displaying the procedure division on-line.
NODMAP	-	Suppress the listing of the COBOL Data division map.
NOPMAP	-	Suppress the listing of the COBOL Procedure map.
NOOFFSET	-	Suppress procedure offset listing (same as NOPMAP).
CLIST	-	Print a condensed listing of the procedure verbs.
NOLIST	-	Suppress program listing (print statistics page only).
DIR	-	List the file directory (assumed when space is low).
VER2	-	Indicates VS COBOL-II or COBOL LE/370 compiler.
CAPEX	-	Indicates use of the CAPEX optimizing compiler.
RETPD=*nnn*	-	Mark for deletion after '*n*' days (the default is 10).
DUMMY	-	Print only. Don't open VSAM file (same as RETPD=Ø).
ENQ=*qname*	-	Override the ENQ "qname" (default is ADSLABEL).
NOVSAM	-	Create intermediate sequential data on SYSUT2 instead of VSAM (for VM/CMS, to update VSAM file remotely).

C (370, MVS, ++, SAS) PROCEDURE

```
//PRE      EXEC CICS translator
//*
//C        EXEC PGM=EDCCOMP,
//              PARM= must include: LIST,NOOFFSET,SHOWINC,RENT,AGGR
                      SAS/C requires: LINENO,SOURCE,OMD,RENT
//SYSCPRT   DD  UNIT=SYSDA,DISP=(MOD,PASS,DELETE),
//              DCB=(RECFM=VBA,LRECL=137,BLKSIZE=882),
//              SPACE=(CYL,(8,3)),DSN=&&PRINT
//SYSUT1    DD  xxxx,DCB=(LRECL=80,BLKSIZE=32ØØ,RECFM=FB)     *
//SYSUT2    DD  xxxx,DCB=(LRECL=80,BLKSIZE=32ØØ,RECFM=FB)
//SYSUT3    DD  xxxx,DCB=(LRECL=80,BLKSIZE=32ØØ,RECFM=FB)
//SYSUT4    DD  xxxx,DCB=(LRECL=80,BLKSIZE=32ØØ,RECFM=FB)
//SYSUT5    DD  xxxx,DCB=(LRECL=80,BLKSIZE=32ØØ,RECFM=FB)
//SYSUT6    DD  xxxx,DCB=(LRECL=80,BLKSIZE=32ØØ,RECFM=FB)
//SYSUT7    DD  xxxx,DCB=(LRECL=3200,BLKSIZE=64ØØ,RECFM=FB)
//SYSUT8    DD  xxxx,DCB=(LRECL=3200,BLKSIZE=64ØØ,RECFM=FB)
//SYSUT9    DD  xxxx,DCB=(LRECL=137,BLKSIZE=685Ø,RECFM=VB)
//*
//ERRS     EXEC PGM=ADSLABEL,PARM='DUMMY,NOPMAP',COND=(5,GT,C)
//SYSPRINT DD  SYSOUT=*
//SYSIN     DD  DSN=&&PRINT,DISP=(OLD,DELETE)
//*
//ADS      EXEC PGM=ADSLABEL,PARM='options',COND=(5,LT,C)
//SYSPRINT DD  SYSOUT=*
//SYSIN     DD  DSN=&&PRINT,DISP=(OLD,DELETE)
//SYSUT1    DD  DISP=SHR,DSN=vsam.adsfile
//*
//PRELINK EXEC PGM=IEWL,COND=(5,LT,C)
```

PARM *options*: (for ADS jobstep)

PGM=&*Id*	-	Program name (default C.SYSIN member name).
NOPMAP	-	Suppress printout of the pseudo assembler listing.
NOLIST	-	Suppress program listing (print statistics page only).
DIR	-	List the file directory (assumed when space is low).
RETPD=*nnn*	-	Mark for deletion after '*n*' days (the default is 1Ø).
DUMMY	-	Print only. Bypass VSAM file (same as RETPD=**Ø**).
ENQ=*qname*	-	Override the ENQ "qname" (default is ADSLABEL).
NOVSAM	-	Create intermediate sequential data on SYSUT2 instead of VSAM (for VM/CMS, to update VSAM file remotely).

***** **D**efault DCB parameters may produce blank pages in the "Pseudo Assembler Listing".

PL/1 PROCEDURE

```
//PRE     EXEC  pre-processor  (Command or Macro Level)
//*
//PLI     EXEC  PGM=IEL0AA,
//              PARM=  must include the following options:
          'OFFSET, STMT, AGGR, ATTRIBUTES(FULL), MAP, STORAGE'
//SYSPRINT DD   UNIT=SYSDA,DISP=(MOD,PASS,DELETE),
//              DCB=(LRECL=121,RECFM=FB,BLKSIZE=2662),
//              SPACE=(121,(25000,500)),DSN=&&LISTING
//*
//ERRS    EXEC  PGM=ADSLABEL,PARM='DUMMY',COND=(5,GT,PLI)
//SYSPRINT DD   SYSOUT=*
//SYSIN     DD  DSN=&&LISTING,DISP=(OLD,DELETE)
//*
//ADS     EXEC  PGM=ADSLABEL,PARM='options',COND=(5,LT,PLI)
//SYSPRINT DD   SYSOUT=*
//SYSIN     DD  DSN=&&LISTING,DISP=(OLD,DELETE)
//SYSUT1    DD  DISP=SHR,DSN=vsam.adsfile
//*
//LKED    EXEC  PGM=IEWL,COND=(5,LT,PLI)
```

PARM *options*: (for ADS jobstep)

`PGM=&Id` - Program Id. (required, there is no default name).

`NOSOURCE` - Generates label reference information only. Saves disk space by not displaying source code on-line.

`NOLIST` - Suppress program listing (print statistics page only).

`DIR` - List the file directory (assumed when space is low).

`RETPD=nnn` - Mark for deletion after '*n*' days (the default is 10).

`DUMMY` - Print only. Don't open VSAM file (same as RETPD=0).

`ENQ=qname` - Override the ENQ "qname" (default is ADSLABEL).

`NOVSAM` - Create intermediate sequential data on SYSUT2 instead of VSAM (for VM/CMS, to update VSAM file remotely).

ASSEMBLER PROCEDURE

```
//PRE     EXEC  pre-processor  (Command or Macro Level)
//*
//ASM     EXEC  PGM=assembler (IFOXØØ, "H", or High-Level)
//SYSPRINT DD   UNIT=SYSDA,DISP=(MOD,PASS,DELETE),
//              DCB=(LRECL=121,RECFM=FB,BLKSIZE=2662),
//              SPACE=(121,(25ØØØ,5ØØ)),DSN=&&LISTING
//*
//ERRS    EXEC  PGM=ADSLABEL,PARM='DUMMY',COND=(5,GT,ASM)
//SYSPRINT DD   SYSOUT=*
//SYSIN     DD  DSN=&&LISTING,DISP=(OLD,DELETE)
//*
//ADS     EXEC  PGM=ADSLABEL,PARM='options',COND=(5,LT,ASM)
//SYSPRINT DD   SYSOUT=*
//SYSIN     DD  DSN=&&LISTING,DISP=(OLD,DELETE)
//SYSUT1    DD  DISP=SHR,DSN=vsam.adsfile
//*
//LKED    EXEC  PGM=IEWL,COND=(5,LT,ASM)
```

PARM *options*: (for ADS jobstep)

PGM=&*Id*	-	Program Id. (default is the name of the first CSECT).
NOSOURCE	-	Generates label reference information only. Saves disk space by not displaying source code on-line.
NOLIST	-	Suppress program listing (print statistics page only).
DIR	-	List the file directory (assumed when space is low).
RETPD=*nnn*	-	Mark for deletion after '*n*' days (the default is 1Ø).
DUMMY	-	Print only. Don't open VSAM file (same as RETPD=Ø).
ENQ=*qname*	-	Override the ENQ "qname" (default is ADSLABEL).
NOVSAM	-	Create intermediate sequential data on SYSUT2 instead of VSAM (for VM/CMS, to update VSAM file remotely).

LINK-EDIT PROCEDURE

Use this procedure only when more than one application program is combined to form a single, composite, load module. It gives you the ability to debug each individual program as if it were defined with its own PPT entry. It is not necessary to use this procedure when the program components are written in COBOL-II or LE and referenced via "static calls".

```
//LKED   EXEC  PGM=IEWL,PARM='LIST,XREF,.....'
```
or
```
//LKED   EXEC  PGM=IEWBLLNK,PARM='LIST,MAP'
```
(MVS 4.1 +)
```
//*
//SYSPRINT DD  UNIT=SYSDA,DISP=(NEW,PASS),SPACE=(TRK,(5,1)),
//             DCB=(RECFM=FB,LRECL=121,BLKSIZE=2662),DSN=&&LIST
//*
//ADS    EXEC  PGM=ADSLABEL
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  DSN=&&LIST,DISP=(OLD,DELETE)
//SYSUT1   DD  DISP=SHR,DSN=vsam.adsfile
```

ALTERNATE (AMBLIST) PROCEDURE

```
//AMBLST EXEC  PGM=AMBLIST
//SYSLIB   DD  DISP=SHR,DSN=user.loadlib
//SYSPRINT DD  UNIT=SYSDA,DISP=(NEW,PASS),
//             SPACE=(TRK,(25,3)),DSN=&&PRINT
//SYSIN    DD  *
  LISTLOAD MEMBER=modname,OUTPUT=XREF
/*
//*
//ADS    EXEC  PGM=ADSLABEL,PARM='options'
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  DSN=&&PRINT,DISP=(OLD,DELETE)
//SYSUT1   DD  DISP=SHR,DSN=vsam.adsfile
```

PARM *options*:

NOLIST - Suppress AMBLIST output (print statistics page only).

VSAM FILE MAINTENANCE

These statistics appear on the last page of your listing:

```
*************************************
*  ADVANCED DEBUGGING SYSTEM  RLS 10  *
*************************************
I N I T I A L    F I L E    S T A T U S
*************************************
     NUMBER OF VSAM DATA RECORDS = nnn
     NUMBER OF EXTENTS ALLOCATED = nnn
     AVAILABLE FREE RECORD SPACE = nnn
     NUMBER OF CNTRL AREA SPLITS = nnn

      STATISTICS FOR PROGRAM xxxxxxxx:
                 nnnn DATA DIV. NAMES.
                 nnnn PARAGRAPH NAMES.
                 nnnn PROCEDURE STMTS.

                 nnn RECORDS REPLACED.
                 nnn RECORDS INSERTED.
```

When free space falls below 4,000 records, the oldest expired program is automatically deleted. You may have to reorganize the file, periodically, to reclaim space that may be wasted by VSAM control areas.

To delete all programs whose retention period has expired:

```
//PURGE  EXEC  PGM=ADSLABEL,PARM='PURGE,RETPD=Ø' **
//SYSPRINT DD  SYSOUT=*
//SYSUT1   DD  DISP=SHR,DSN=vsam.adsfile
//SYSIN    DD  DUMMY
```

****** Use RETPD= to specify number of days to force an earlier retention period.

To manually force deletion of specific programs, regardless of their expiration dates:

```
//PURGE  EXEC  PGM=ADSLABEL
//SYSPRINT DD  SYSOUT=*
//SYSUT1   DD  DISP=SHR,DSN=vsam.adsfile
//SYSIN    DD  *
PURGE
progname1
progname2
   " "
/*
```

Considerations for DFHSRP

During ADS startup, the program DSPINITL applies a dynamic patch to DFHSRP.

This following *ADS* message may be issued at 'sign-on':

UNABLE TO INITIALIZE PGM=DFHSRP

This indicates that DFHSRP is protected (in LPA) and cannot be initialized by ADS.
The solution is to remove DFHSRP from the LPA (by specifying `'PRVMOD=DFHSRP'` in the CICS JCL parm and `SHR=NO` in the NLT). You may have to relink-edit DFHSRP with the `NORENT` option if necessary.

VERIFY GLOBAL MONITORING

Use the following on-line procedure to verify that *ADS* will monitor programs "by terminal":

Activate terminal monitoring:	`ADSP READY TRACE`
The response should indicate:	`*TRACE TERMINAL IS IDLE`
Hit the ENTER key again:	`*TRACEØ TERMINAL IS IDLE`
Remove terminal monitoring:	`ADSP DELETE TRACE`

The number zero is appended to the `*TRACE` indicator by the *ADS* global exit (`DSPEXITP`). It indicates that *ADS* is active at this terminal and shows how many programs were monitored. If a numeric digit is not displayed, after hitting ENTER the second time, it means that the *ADS* global exit is not enabled. Look for, and correct, the possible causes:

a. A PPT entry is missing for either DSPEXITP or DSPROUTE

   Action: Add the PPT entry (or CSD) and use CECI to *enable* the global exit:

   ```
   CECI ENABLE PRO(DSPEXITP) EXIT(XPCFTCH) START
   ```

b. The SIT is not generated to allow GLOBAL exits.

   Action: Specify "EXITS=YES" either in the SIT or the CICS parm.

c. The ADSM/ADSP transactions may not be authorized to enable the exit.

   Action: Remove the security restriction from the PCT.

The global exit program (`DSPEXITP`) is distributed as a source macro. It lets you identify the generic ("wildcard") names of programs that should not be monitored automatically via the "READY TRACE" command. It does not prevent you from using *ADS,* on such programs, to set breakpoints or to monitor them with any of the specific "ASSIGN MONITOR" functions. It is important, therefore, to define all of your "system-type" packages and non-standard applications.